This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2010 proceedings

Mesh-based Broadband Home Network Solution: Setup and Experiments

Djamal-Eddine Meddour, Abdesselem Kortebi
Orange Labs
2, Avenue Pierre Marzin, 22300 Lannion, France
{djamal.meddour, abdesselem.kortebi}@orange-ftgroup.com

Raouf Boutaba
School of Computer Science
University of waterloo
Waterloo, ON N2L 3G1, Canada
rboutaba@uwaterloo.ca

***Abstract*— In this paper, we investigate architectural and practical issues related to the setup of a broadband home network solution. Our experience led us to the consideration of a hybrid, wireless and wired, Mesh-Network to enable high data rate service delivery everywhere in the home. We demonstrate the effectiveness of our proposal using a real experimental testbed. This latter consists of a multi-hop mesh network composed of a home gateway and "extenders" supporting several types of physical connectivity including PLC, WiFi, and Ethernet. The solution also includes a layer 2 implementation of the OLSR protocol for path selection. We developed an extension of this protocol for QoS assurance and to enable the proper execution of existing services. We have also implemented a fast WiFi handover algorithm to ensure service continuity in case of user mobility among the extenders inside the home.**

Keywords-component; Home Network, Distributed Architecture, Testbed, Services

I. INTRODUCTION

During the recent years**,** home networks constituted a new territory of opportunity for network operators. Several fora have been established to cover different home networking issues such as device interoperability and home gateway functions: Home Gateway Initiative (HGI) [1], Universal Plug and Play (UPnP) [2], Digital Living Network Alliance (DLNA) [3].

In this environment, the consumers expect to enjoy a variety of services such as multimedia streaming, files sharing, gaming, Internet browsing and High Definition TV with a high quality of experience. However, many challenges need to be addressed first such as QoS assurance and user-friendliness. On the other hand, the emergence of new multimedia devices and the increasing availability of bandwidth in access networks allow for a full coverage in the home through a more complex network infrastructure. This raises the need for the definition of new and tailored network architectures.

Using an experimental setup, this work explores a novel home network architecture based on the introduction of additional network elements called *extenders* which implement a path selection at layer 2 with a converged multi-hop connectivity of various link technologies. We propose an extension of this protocol for QoS assurance and for the proper support of existing services. Furthermore and to deal with the multihop nature of the envisioned home network solution, we implement a specific WiFi handover algorithm.

This paper is organized as follows. In the next section, we introduce a new distributed architecture for emerging home networks. Section III presents the implemented path selection protocol. The testbed configuration is described in section IV; where the conducted experiments are described together with the fast WiFi handover algorithm implementation. Finally, we conclude in section V.

II. HYBRID MESH HOME NETWORK ARCHITECTURE

The increase of network access rates especially with more and more deployment of FTTH (Fiber To The Home), the proliferation of terminals and consumer devices inside the home, as well as the creation of new services together contribute to an increasing demand for a new full-fledge home network architecture. Indeed, the current multiple play home networks (now offering Internet, VoIP, mobile telephony and IPTV services) are home gateway (HGw) centric. The penetration of ultra broadband throughput to the entire house implies the extension of physical connectivity into all living spaces. Consequently, the new home network architecture should be distributed since the HGw centric model may lead to a bottleneck in addition to coverage concerns. Our proposed architecture is based on the introduction of new equipments called *extenders* [4], [5] which connect to the home gateway and to the user devices. The extenders are multi technology devices including several interfaces of different types such as PLC (Power Line Communication), WiFi, and Ethernet. Their

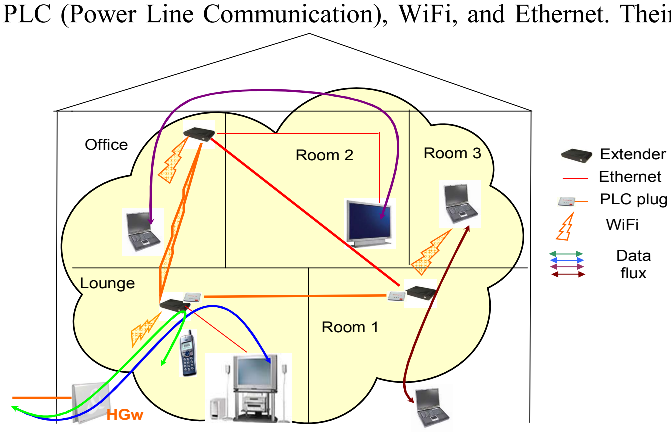

Figure 1. Hybrid mesh home network

main function is to extend the coverage and improve the home network performance by spreading the connectivity. The various link layer technologies offer different tradeoffs between throughput, coverage and ease of deployment. Furthermore, the interconnection of the extenders naturally forms a mesh network allowing for multiple paths between a source and a destination. A path selection protocol is hence needed to select the best available path for the flows (both intra-LAN and WAN-LAN traffic). Such a protocol must be efficient, simple and robust. Fig. 1 illustrates a distributed architecture forming a hybrid mesh-based home network. This architecture provides the flexibility to perform load balancing, and to deal with link quality degradation and/or failures inside the home network. The goals underlying the proposed architecture are to provide an easy to deploy, low cost and reliable home network.

III. PATH SELECTION

Since we are considering a multi-hop mesh network, a routing protocol shall be used. However, we use the terminology of path selection instead of routing because the protocol operates at layer 2 of the OSI model. This choice is basically motivated by the need to keep the home network as a single IP sub network for user convenience and simplicity design reasons. In fact, the extenders have to be plug and play equipments avoiding any user intervention for configuration purposes. Furthermore, broadcast messages inside the home network have to reach all devices

A. *Protocol descritption*

Usually, the spanning tree protocol (STP) [6] is the advocated protocol when operating at layer 2. However, STP presents many drawbacks: network capacity underutilization (some links are blocked to avoid loops) and path reconfiguration takes into account failure events but not congestion. The intent is to overcome these shortcomings as described later.

The path selection is performed by means of an extended version of AWDS (Ad-hoc Wireless Distribution Service) protocol [7]. It is an OLSR (Optimized Link State Routing) [8] implementation operating at L2 (MAC layer). It combines the simplicity of layer 2 path selection with the self-* properties of a MANET routing protocol. It is worth mentioning that the basic AWDS implementation handles only one wireless interface per node. It is implemented as a Unix/Linux daemon and operates completely in the user space.

We enrich the path selection protocol with the following features

B. *Multiple interfaces support*

As previously noted, heterogeneous communication technology is expected. Therefore, supporting multiple interfaces at the path selection level is essential. Since the protocol makes abstraction of the interface type, our extension is simple as indicated in Fig. 2. A queue is created for each existing interface, and is connected to a daemon. Therefore, it will be able to receive, send signaling messages over all

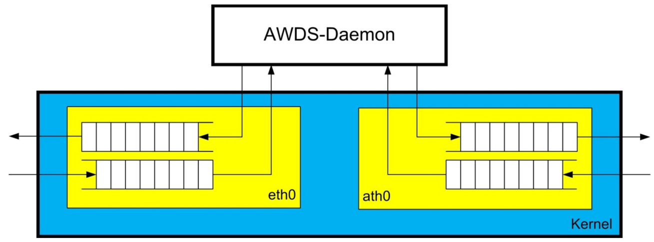

Figure 2. AWDS, multiple (2) interfaces architecture

existing interfaces; calculate paths and forward data packets accordingly.

C. *QoS support*

For the path selection criterion, the protocol relies on the ETX (Expected Transmission Count) metric. It is defined as the predicted number of data transmissions required to send a packet over a given link, including retransmissions [9]. This metric proves interesting compared to a simple shortest path scheme. The main drawback of ETX is that it is not able to reflect properly the lost at the application level. In fact, control packets loss ratio is used to detect link quality degradation. These packets have a small size and are usually transmitted with high priority; therefore they will be likely received even in case of congested links.

To overcome these limitations, we extend the current implementation to consider more accurately the link load. We use higher control packets sending rate (10 beacon packets/s and 2 TC (topology) packets/s) and we transmit a fraction of control packets with the same priority as the data packets. The algorithm is the following (performed at each node):

- *Send beacon packets (all neighbours) with two priority levels*
- *On the reception of beacon packets, construct the local topology; compute the ETX metric value over each link in the neighbourhood*
- *Send the TC (topology control) packets to disseminate the local topology to the other nodes in the network.*
- *On the reception of TC messages, update the global network topology*
- *Compute paths*

IV. EXPERIMENTAL SETUP

We conducted an experimental study by setting up a real testbed for proof of concept and performance evaluation purposes. In this section, we present its configuration. Then an AWDS evaluation is computed. Finally, a fast WiFi handover solution is described.

A. *Hardware and softeware configuration*

The testbed, as indicated in Fig. 3, is composed of 3 mini PCs running GNU/Linux OS, playing the role of extenders.

The main characteristics of these mini PCs are the following: CPU: VIA C7 1GHz and memory: 1Go. It also includes a home gateway (HGw) which is the interface to the WAN network and constitutes the local DHCP server. The interconnection of the 3 extenders forms a hybrid mesh backbone. As a first step, we considered mature and well established connectivity technologies. Indeed, the extender E1 is connected to E2 through a POF (Plastic Optical Fiber) link of 100Mb/s. The extender E2 is connected to E3 using a PLC

This full text paper was peer reviewed at the direction of IEEE Communications Society subject matter experts for publication in the IEEE ICC 2010 proceedings

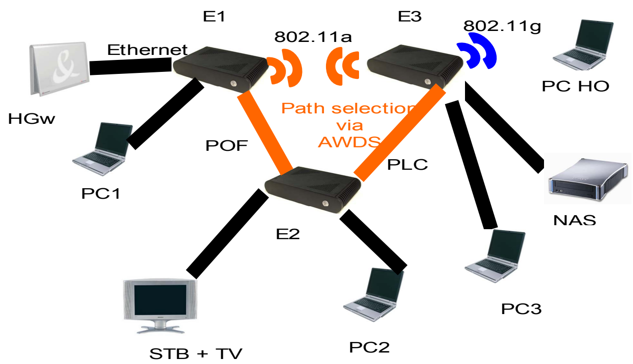

Figure 3. Testbed configuration

(Power Line Communication) segment based on 2 HD live plugs [10] offering a physical capacity of 200Mb/s. Finally, E1 and E3 communicate through an IEEE 802.11a link. Thus, the backbone contains WiFi, PLC and POF links. The user end devices are represented mainly by PCs. More specifically, 4 PCs are used; 3 of them are connected to each one of the extenders through Ethernet links and the last one is associated to one of the extenders using an IEEE 802.11g connection to avoid interferences with the backbone link. This PC is used to illustrate the WiFi handover solution implemented on the testbed (cf. section C). Wireless cards use the well known MadWiFi Linux driver. Furthermore, a TV set along with a STB (Set Top Box) are connected to E2; while an external hard drive (NAS) is connected to E3 for storage purposes.

B. Path selection protocol evaluation

To ensure backward compatibility with legacy devices, we need to run the protocol only on the backbone nodes (extenders). In other words, end devices do not implement AWDS. To connect an end device with an AWDS node (extender), we used the Linux bridge utility thanks to *bridge-utils* package. More specifically the Linux bridge was used to connect the virtual interface (*awds0*) with the access interfaces at each extender. In fact, the Linux bridge is a software that emulates the behavior of a hardware switch. In the following sub-sections, we describe the various conducted experiments.

1) Resource consumption: The first set of experiments was to determine AWDS CPU and memory occupations. Fig. 4 depicts CPU consumption; it is lower than 2% on all three extenders (without data traffic). On the other hand, memory consumption is stable and constant around 3M bytes. Additional experiments showed that CPU consumption depends on network load, while memory occupation remains unchanged.

2) Load balancing experiments: The second type of experiments is to illustrate the full utilization of the mesh network. In fact, AWDS is derived from OLSR as mentioned in section III.A; therefore it does not require the blocking of some links unlike STP and its variants. The conducted experiments show that when 3 flows are generated, they use the 3 backbone links inducing a load balancing at the flow level. More specifically, we tested the following scenario:

- Flow1: file transfer from the PC1 to NAS. It uses the wireless link between E1 and E3.
- Flow2: local video streaming from PC1 (server) to PC2 (client) using VLC player. It passes through the POF link between E1 and E2.
- Flow3: local video streaming from PC2 to PC3 with VLC player. It uses the PLC link between E2 and E3.

Thus all 3 backbone links are used simultaneously. Other types of flows were also generated including web browsing, artificial *iperf* connections and IPTV.

As we can see, AWDS selects the direct path between any pair of extenders to reach a given destination. Clearly the shortest path is not necessarily the best one. However, the used ETX metric privileges the single hop path over the two hops path in normal load situation where there is not congestion; because it relies on control packets (beacon) observed loss rate.

In the future, it would be possible to enrich the path selection metric by taking into account additional QoS parameters (link load, available bandwidth, delays, etc.) to select systematically the best path.

3) Fault tolerance experiments: We conducted experiments to highlight the robustness of the mesh home network based on the AWDS protocol in case of link failures. The intent is to measure the corresponding recovery time when the protocol detects the failure and selects an alternate path to carry the flows. Particularly:

- When the cable between E1 and E2 is unplugged, the flow from PC1 to PC2 is routed over the E1-E3-E2 path (wireless then PLC links) instead of E1-E2. The

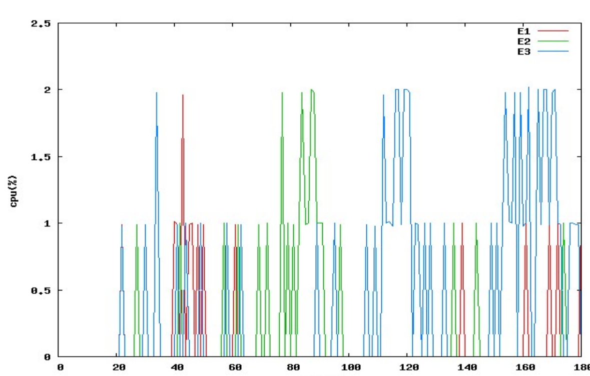

Figure 4. Path selection protocol CPU consumption

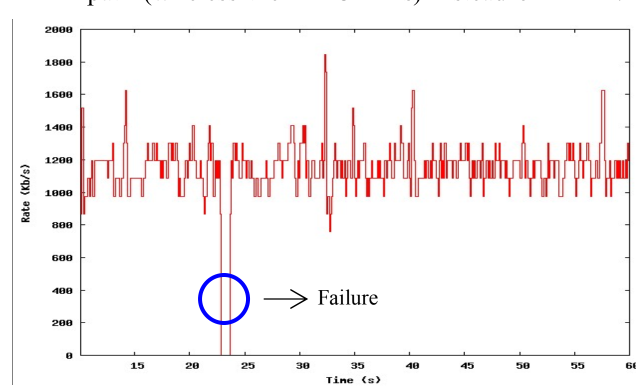

Figure 5. Link failure recovery with real time video traffic

mean measured recovery time is 1s.

- When the cable between E2 and E3 is unplugged, the flow from PC2 to PC3 passes through the path E2-E1-E3 (POF then wireless links) instead of the direct one (E2-E3). The reaction time is 1s.

If the corresponding flow is a video streaming using the UDP protocol, we observe a slight interruption before it resumes quickly. When the single hop link is up again, the flow is routed through it without disruption (there is no visible effect on the video for example). Fig. 5 shows the rate variation of a video flow from PC1 to PC2 during 1 minute. The failure event occurs at 23s instant where the rate drops to 0, the video resumes at 24s instant.

The obtained recovery times are similar to those of RSTP (Rapid STP) [11] according to our lab tests. Nevertheless, they can be further reduced. Indeed, link failures detection is based on the protocol control messages exchange: beacon (for neighbor detection) and topo (for topology discovery) packets which are sent periodically. The corresponding intervals are 100ms and 400ms respectively. The resulting overhead is 10.8Kb/s per link. Naturally, the decrease of these values conducts to faster failure detection at the cost of a higher overhead. A good compromise is therefore suitable.

4) Congestion reaction experiments: The most interesting use case in home networks is probably the reaction to link congestion or overload. In particular, using shared media such as WiFi and PLC with variable bandwidth, flows contention may lead to QoS degradation. The user should be able to enjoy his services even when some segments are saturated, thanks to the use of alternate paths in the mesh network. Therefore, we enhanced the basic version of AWDS by adding the congestion detection support for wireless links.

More specifically, the protocol selects the alternate path when congestion is detected in the WiFi link relying on the quality indication calculated based on the modified ETX metric. Hence, a new and better path is used for all flows passing through the wireless link and the forwarding table is changed accordingly.

We performed experiments using a video streaming flow from PC1 to PC3. The video is transmitted over the wireless link between E1 and E3. Then, we generate a second flow from

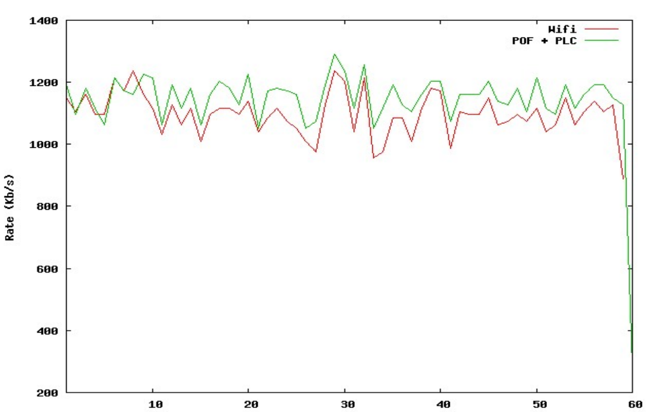

Figure 6. WiFi congestion reaction with real time video traffic

PC1 to PC3 that saturates the WiFi link using the *iperf* tool (a UDP connection of 25Mb/s). The protocol is able to detect the congestion and both flows are switched to POF and PLC links (alternate path) quickly; most of the time there is no perceptible degradation of the video maintaining then a satisfactory QoS as indicated in Fig. 6 green graph (POF + PLC) where the *iperf* flow is started at instant 10s. To illustrate the video degradation where no rerouting is performed, we plot in red graph the video rate variations on the WiFi link. Clearly, AWDS avoids video packets loss when it switches to the alternate path since the obtained rate is higher.

The current implementation of the reaction to congestion needs clearly to be extended to deal with all media types, in addition to WiFi. Furthermore, a good available bandwidth estimation method is required to have better information about network resources.

5) MTU issue: As indicated in section IV.B, when an extender receives a packet from its access interface (the interface connected to an end device), the Linux bridge forwards it to the awds0 virtual interface, that operates the AWDS protocol, and then to the right backbone interface. The protocol encapsulates the packet by adding its own header of 37 bytes. Therefore, when the packet size is greater than (MTU – 37B) value, the maximum authorized packet size is exceeded leading to an incorrect transmission in the backbone network.

According to the different experiments conducted, this issue arose particularly for TCP flows (web browsing and file transfer applications); the associated flows attempt to send at the maximum available rate, and the packet size is usually 1500B. On the other hand, the tested UDP flows (IPTV, video streaming) did not suffer from this problem since their packet size is less than 1463B (MTU – 37B).

One possible solution is to modify the MTU value at each end device. This is easy to perform on the testbed and it works fine. However, it implies a modification of the user terminals which is unsuitable. Therefore, we adopted another solution based on the modification of the MSS (Maximum Segment Size) value at each extender. It is performed through a simple *iptable* rule to bind the MSS to 1423B (1463 – 40). Obviously, it concerns only TCP connections. Nevertheless, it is better than the fragmentation of packets at the extenders. An additional treatment for UDP packets larger than 1463B could consist in sending a message from the extenders to end devices suggesting a decrease of their respective MTU.

C. User mobility: WiFi handover experiments

The presence of multiple extenders inside the home and thus several wireless APs (access points) offers the possibility to provide service continuity for users on the move (between different rooms or even from a floor to another). Their end devices can be attached to the best AP (according to various criteria such as signal strength, load or others). The handover must be realized seamlessly to avoid sessions' interruption. We implemented a fast WiFi handover algorithm in our testbed using the MadWiFi driver (version 0.9.4) on PC HO of Fig 3. The proposed solution is based on selective active scanning and a cache mechanism to accelerate the handover time. It is known that the handover delay can be divided into three parts: scanning (AP discovery), re-authentication and re-association. The scanning phase constitutes almost 90% of the total delay [12]. Thus, it has to be fast.

The current implementation uses the RSSI (Received Signal Strength Indication) as the only indicator for the handover. Periodically, the driver checks the RSSI level; if it is below a certain threshold, the terminal looks for a better AP. In this perspective, a cache table is used and contains all APs detected by the end device and their corresponding RSSI. If the cache is valid, the terminal tries to associate to the best AP (if it is different than the current one). Otherwise, if the cache is too old; an active scanning is performed to refresh the cache. Note that, several parameters have been tuned to decrease the scanning duration: minimum and maximum probe intervals, number of scanned channels (focus on IEEE 802.11g and starting with the most used channels) and number of probe requests. The following pseudo code summarizes the handover algorithm actions:

Every HO_period
    if *(RSSI < HO_threshold)*
        Check APs cache table
        if *(cache too old)*
            Perform enhanced selective active scanning
            Update APs cache table
            Associate to the best AP
            Send ARPING message
        else
            Associate to the best AP
            Send ARPING message

The conducted experiments show that when the handover is performed, the degradation of the video streamed from PC1 to PC HO using VLC player and UDP protocol disappears quickly indicating that the handover is fast. Furthermore, the perturbation duration includes the rerouting delay in addition to the handover time itself, since the end to end path between PC1 and PC HO changes when PC HO is associated with the new extender. To accelerate new path calculation, we generate an ARPING message from the end device to the HGw after the re-association phase. We also conducted another test with a VoIP flow using Skype Soft phone from PC HO to an external machine (in the Internet); when the handover is performed, the communication is maintained and the degradation is hardly perceivable. Therefore, seamless handover is ensured.

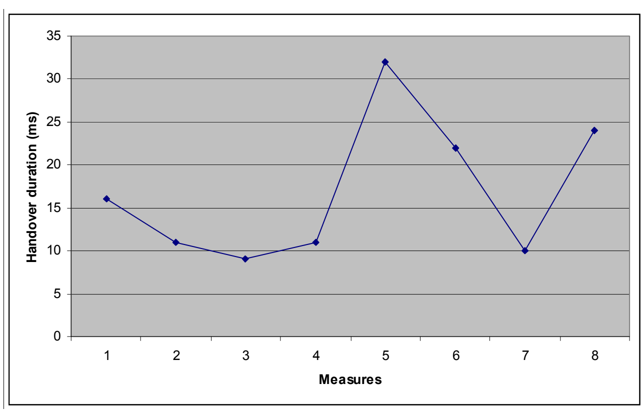

Figure 7. Handover duration measurements using ping messages

More specifically, the *Wireshark* analyzer indicates very low average handover duration of 15ms as indicated in Fig. 7 including rerouting time, using ping messages (1 ICMP packet/ms) from PC HO to PC1. When we performed a second series of tests with an *iperf* UDP flow at a rate of 11.6Mb/s (1 packet/ms), the resulted average duration was 82.1ms. In fact, the *ping* messages sent in the first set of experiments seem to accelerate the computation of the new path after the handover. Note that these results were obtained using a WEP based authentication mechanism.

V. CONCLUSION

In this paper, we designed and evaluated using a real testbed a distributed home network architecture formed by a hybrid mesh network based on extender devices. The latter bridges heterogeneous link layer technologies transparently and expand the coverage to the entire home. Our solution includes a path selection protocol extension and a fast WiFi handover algorithm in order to set up the above-mentioned features. Indeed, the path selection protocol ensures load balancing, robustness and enhances QoS in case of wireless links congestion. On the other hand, the implemented fast WiFi handover solution preserves VoIP sessions continuity and limits video degradations in case of user mobility inside the home according to our conducted experiments.

In our future work, we plan to enrich the path selection protocol with more capabilities; particularly: per flow path selection, link monitoring and management. Indeed, load balancing would be then more efficient with per flow routing rather than source-destination based. Moreover, we target a better handling of multicast flows avoiding packets flooding.

REFERENCES

[1] Home Gateway Technical Requirements: Residential Profile. HGI-RD001-R2.01. 2008

[2] Universal Plug and Play [Online]. http://www.upnp.org/

[3] Digital Living Network Alliance [Online]. http://www.dlna.org/home

[4] D. E. Meddour et al., "Technologies for the Wireless Future: Wireless World Research Forum", WWRF BoV Chapter 10, Volume 3, September 2008, ISBN: 978-0-470-99387-3

[5] V. Freiderikos, et al., "Piloting Home Network with Intelligent Agents". WWRF, Stockholm, 2008

[6] ANSI/IEEE 802.1D-2004 standard, MAC Bridges.

[7] AWDS project [Online]. http://awds.berlios.de

[8] T. Clausen, P. Jacquet, Optimized Link State Routing protocol. RFC 3626. October 2003.

[9] Douglas. De Couto, D. Aguayo, J. Bicket, R. Morris. "A high-throughput path metric for multi-hop wireless routing", In *Proc. ACM/IEEE MobiCom*, September 2003.

[10] http://www.homeplug.org

[11] ANSI/IEEE 802.1W-2004 standard, Rapid Reconfiguration of Spanning Tree

[12] A. Mishra, M. Shin, and W. Arbaugh, "An empirical analysis of the ieee 802.11 mac layer handoff process", *SIGCOMM*, 2003